Identifying Accident Causes of Driver-Vehicle Interactions Using System Theoretic Process Analysis (STPA)

Shufeng Chen
WMG, University of Warwick
Coventry, United Kingdom
Shufeng.Chen@warwick.ac.uk

Siddartha Khastgir
WMG, University of Warwick
Coventry, United Kingdom
S.Khastgir.1@warwick.ac.uk

Islam Babaev
Arrival
London, United Kingdom
babaev@arrival.com

Paul Jennings
WMG, University of Warwick
Coventry, United Kingdom
Paul.Jennings@warwick.ac.uk

***Abstract*—Latest generations of automobiles are gradually being equipped with technologies that have increasing automation, a trend which had led to increase in the system complexity as well as increased human-automation interactions. Failures in such complex human-automation interactions increasingly occur due to the mismatch between what operators know about the system and what the designers expect operators to know. Causes of road accidents also change due to role shift of drivers from controlling the vehicle to monitoring the in-vehicle controllers. Failures in such complex systems involving human-automation interactions increasingly occur due to the emergent behavior due to the interactions, and are less likely due to reliability of individual components. Traditional safety analysis methods fall short in identifying such emergent failures. This paper focuses on using a systems thinking inspired safety analysis method called System Theoretic Process Analysis (STPA) to identify potential failures. The analysis focuses on a SAE Level-4 Vehicle that is controlled partially by a safety driver and its built-in Autonomous Driving System (ADS). The analysis yields that while increase in complexity does increase system functionality, it also brings a challenge to evaluate the safety of the system and potentially causes incorrect human-automation interactions, leading to an accident. After the possible inadequate driver-vehicle interactions are identified by STPA, corresponding requirements were then proposed in order to avoid the unsafe behavior and thus preventing the hazards.**

Keywords—STPA, systems thinking, human-automation interaction

I. INTRODUCTION

Over the last many decades, automotive systems have become increasingly complex, with the current luxury car having over 100 million lines of code [1]. The introduction of automation through various Advanced Driver Assistance Systems (ADAS) and Autonomous Driving Systems (ADS) have led to a significant challenge in terms of safety evaluation [2]. As the complexity of in-vehicle human-automation interaction has increased, so has the number of operational modes and the number of ways of triggering those operational modes. However, not only will unsafe system potentially lead to an accident, it also has a potential to lead to lack of trust in the systems [3]. Safety of a vehicle is usually linked to the reliability of the components within the vehicle. However, with technological developments in the last decades, component designs have become mature and possibility of individual component failure has dropped significantly due to their increased reliability. Failures in complex systems such as ADAS and ADS tend to occur due to emergent behaviour resulting from inadequate subsystem interactions. When a vehicle without any component failures is involved in an accident, the drivers are usually blamed as the causal factor, although they have performed as expected or as would be reasonable. One of the similar accidents reflecting this situation was the recent Tesla Autopilot crash [4]. The Tesla driver was blamed due to his disengagement to the vehicle control, however, the driver's mental model was trained to believe that Autopilot system was capable of handling the situation, suggesting over-trust in the system, and lack of informed safety.

Statistically, it is suggested that to prove ADS systems are safer from human-driven vehicles, they need to be driven for over 11 billion miles [5]. As this might seem reasonable, for complex systems, safety evaluation has had a shift from understanding "how a system works" to "how a system fails" [6], with a focus on quality of miles – "smart miles". Considering Rumsfeld's Known and Unknown Metrix [22], there are four quadrants representing different categories of hazards leading to accidents. The quadrant that represents unknown hazards has now become the challenges in the domain of safety.

In the past few decades, many hazard identification methods have been used to identify "how a system fails", including Failure Mode and Effects Analysis (FMEA) [7], [8], Fault Tree Analysis (FTA) [9], [10], Event Tree Analysis (ETA), Hazard and Operability Analysis (HAZOP) [11] etc. Most of the system failures identified by these methods are due to the inconsistencies between system performance and system requirements (assuming that the requirements are always correct) – i.e. they are known hazards. However, when the requirements are inadequate or become less adequate over time, the potential hazards of a system become unknown, leading to a large area with unknown hazards. A diverse range of causal factors of system failures therefore need to be considered.

Whilst considering accidents in a human-automation system, there has been a variety of analysis methods developed since

Manuscript 870 submitted to 2020 IEEE International Conference on Systems, Man and Cybernetics (SMC). Received May 29, 2020.

1990s in order to identify causes of human errors in a socio-technical system, including AcciMap Approach [12], Functional Resonance Analysis Method (FRAM) [13], Human Factors Analysis and Classification System (HFACS) [14], HERA-JANUS [15] .etc. These methods commonly illustrate the diversity of causal factors across different levels of the systems, their interactions and the roles played by external influences such as political, cultural, financial and technical circumstances. When analysing causes of accidents using these methods, they are either based on retrospective accidents or are elaborate, requiring contributions from different teams. This also brings challenges when the deliverables of analysis are needed in order to facilitate the system development in a fast-paced engineering life cycle. Therefore, a new method is needed in order to identify a diverse range of prospective hazards efficiently.

Contrary to the hazard identification methods mentioned earlier which consider accidents as a chain of events, systems thinking inspired System Theoretic Process Analysis (STPA) believes that they occur most likely when external disturbances or dysfunctional interactions among system components are not adequately handled by the control system [16]. STPA can potentially identify causes which may not be identified by other methods, especially those concerning human-computer interaction, software bugs, missing requirements and even socio-technical factors and it prevents accidents by enforcing constraints on component behaviour requirements and interactions. While STPA was originally applied in space applications [17], more recently it has had wide applications across aviation, automotive, medical, defence and nuclear industries.

II. STPA Methodology

A. *STPA Step 1: Define Purposes of the Analysis*

As a top-down approach, STPA starts by identifying any unacceptable losses, including loss of human life or human injury. STPA specified losses may not be limited to safety-critical losses. For example, environmental pollution, loss of mission, loss of reputation may not be safety-critical, but they are also treated as losses in STPA because they are unacceptable to the stakeholders.

The system boundary is then determined. System boundary defines the range of controllability – i.e. any components outside the system boundary are not controllable to the designer and any components within the boundary can be controlled or redesigned. In STPA, if a vehicle under analysis involves human control, the corresponding human drivers or operators are also treated as components of the system as they can be trained or guided to control the vehicle in a pre-determined manner.

Once unacceptable losses have been determined, system-level hazards are then identified. System-level hazards describes a system state or set of conditions that lead to a loss at a particular set of worst-case environment conditions.

B. *STPA Step 2: Model the Control Structure*

The aim of this step is to create a hierarchical control structure that is composed of nested feedback control loops between sub-systems. A generic feedback control loop is illustrated in Fig. 1. In general, a controller may provide control actions (CA) to control some process and to enforce constraints on the behaviour of the controlled process. The control

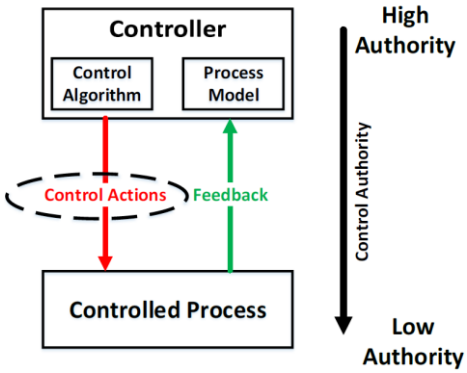

Fig. 1 Generic Feedback Loop

algorithm represents the controller's decision-making process – i.e. it determines the CA to be provided. Controllers also have process models that represent the controller's internal beliefs and assumptions (i.e. its view of the outer world) used to make decisions. Process models may include beliefs about the process being controlled or other relevant aspects of the system or the environment. Process models may be updated in part by feedback used to observe the controlled process. For example, a driver provides 'brake' CA to the vehicle based on the 'vehicle speed' feedback from Human Machine Interface (HMI). Here, driver is telling the vehicle to slow down and HMI which is part of the vehicle is updating the driver with the current vehicle speed.

C. *STPA Step 3: Identify Unsafe Control Actions (UCA)*

After identifying CA in the control structure, each CA is further analysed to identify how the CA would manifest into a UCA. In a certain circumstance, a correct CA could lead to one or multiple system-level hazards (as identified in Step 1). If a CA were always unsafe, then it would never be included in the system design. To identify a UCA, the CA is usually considered together with a particular context and there are several guidewords that can be used to identify UCAs:

- **Not providing** the CA leads to a hazard
- **Providing** the CA **incorrectly** or **when not needed** leads to a hazard
- **Providing** the CA **too early** or **too late** or **in the wrong order** leads to a hazard
- **Providing** the CA **too long** or **stopped providing** the CA **too soon** leads to a hazard

D. *STPA Step 4: Identify Loss Scenarios and Requirements*

Once the UCAs are identified, each UCA is further analysed to identify the possible loss scenarios of the UCA. Loss scenarios are usually considered as the combinations of the process model belief, reasons for the belief and the corresponding causal factors, with the aid of the specific control loop of that CA as illustrated in Fig. 2. For a UCA to occur, the process model of the controller has a belief based on which it believes that the CA it is directing is safe when it is actually unsafe [18]. Decisions made by the process model could inevitably be incorrect if its belief is inadequate. In the meantime, the accuracy of the process model belief is significantly determined by the inputs it receives.

Manuscript 870 submitted to 2020 IEEE International Conference on Systems, Man and Cybernetics (SMC). Received May 29, 2020.

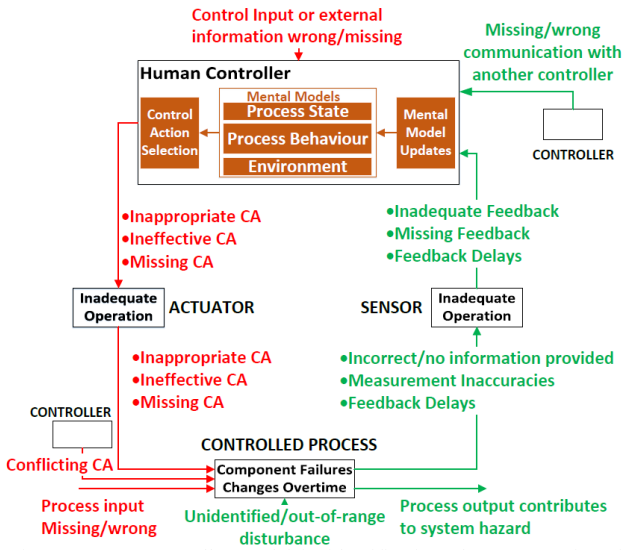

Fig. 2 Human Controller Model for identification of Loss Scenarios of human operators[19]

Whilst analysing loss scenarios of UCAs from a Human Controller (e.g. Driver, Operator), the human mental model is applied. Unlike software decisions made by process model, the way humans make decisions depends on a tremendous amount on the context in which a particular decision is made [19]. The goal of applying STPA on human controller is not just to understand how the human operators think, but also to understand how and why they violate the safety constraints of the system. Fig. 2 also shows the human controller with three stages that includes receiving and processing information and selecting control actions. In order to identify the three stages, three questions need to be answered:

- How did the operator choose which control action to perform?
- What does the operator know or believe about the system?
- How did the operator come to have their current knowledge or beliefs?

For each causal factor identified, corresponding requirements are proposed either to prevent the causal factor or to enable the system to detect the causal factor [20]. Ideally, the causal factor shall be prevented as a top priority in order to avoid the UCA. However, it is also possible that the causal factor cannot be prevented and therefore, the back-up requirement is proposed so that the causal factor can be detected or exposed.

III. STPA on Safety Driver-Vehicle Interactions

Whilst there are nearly 40 UCAs from a safety driver, two example UCAs are analysed in this section in order to identify the diversity of causal factors of accidents involved in driver-vehicle interactions.

A. *STPA Step 1: Define Purposes of the Analysis*

To start with, a list of losses are identified as presented in TABLE I. It is worth noting that neither L-3 nor L-4 in TABLE I are safety-critical losses, but they are indeed unacceptable to the stakeholders.

TABLE I. A List of Losses

	Losses
L-1	Loss of life or injury to drivers, passengers or pedestrians
L-2	Loss of damage to the vehicle or objects outside the vehicle
L-3	Loss of transportation mission
L-4	Loss of customer satisfaction or confidence on autonomous vehicle

The system boundary in this analysis includes the Autonomous Vehicle and the safety driver. A list of vehicle-level hazards are identified in TABLE II. It is important to note that the hazards here represent the system-level state, and therefore the conditions of subsystems are not considered in this step. Each system-level hazard could trigger more than one losses. For example, for H-1, collision with a pedestrian could cause death of the pedestrian or injury to the passenger (as per L-1), and vehicle might also be damaged (as per L-2). Inevitably, the transportation mission is terminated due to the accident (as per L-3). Consequently, the corresponding vehicle company will lose satisfactions and confidence from customers (as per L-4).

TABLE II. A List of Vehicle-Level Hazards

Vehicle-Level Hazards		Link to Losses
H-1	Collision with pedestrians, animals or other road users	L-1,2,3,4
H-2	Vehicle fails to follow pre-defined route	L-3,4
H-3	Vehicle fails to follow road structures (roundabouts, junctions .etc.)	L-3,4
H-4	Vehicle fails to follow traffic rules	L-3,4

B. *STPA Step 2: Model the Control Structure*

At the initial iteration of the analysis, a very high level of control structure was created as illustrated in Fig. 3. The control structure includes ADS, Brake-by-Wire system, sensors and actuators. At this abstraction level of control structure, the subsystems are considered as black boxes and therefore only the interactions among the black boxes as well as the behaviour of each black box are analysed. The more detailed components inside each boxes are explored in the next iteration of the analysis.

C. *STPA Step 3: Identify Unsafe Control Actions (UCA)*

Some of the UCAs from Safety Driver are captured in TABLE III. For demonstration purposes, two example UCAs from Safety Driver are discussed in this paper.

TABLE III. A List of Losses

CA	Press Brake Pedal
Not Provided	UCA-1: Safety Driver does not press brake pedal when current speed is higher than nominal speed and ADS is disabled (H-4)
Provided incorrectly/when not needed	UCA-2: Safety Driver presses brake pedal with insufficient amount when current speed is still too high for the sharp turn ahead. (H-2,3)
Provided too early/too late	UCA-5: Safety Driver pressed brake pedal too late when vehicle speed is not safe for the sharp turn ahead and ADS is disabled (H-2,3)
Provided too long/stopped providing too soon	UCA-7: Safety Driver stopped pressing brake pedal too soon when vehicle speed is still higher than nominal speed and ADS is disabled (H-4)

Manuscript 870 submitted to 2020 IEEE International Conference on Systems, Man and Cybernetics (SMC). Received May 29, 2020.

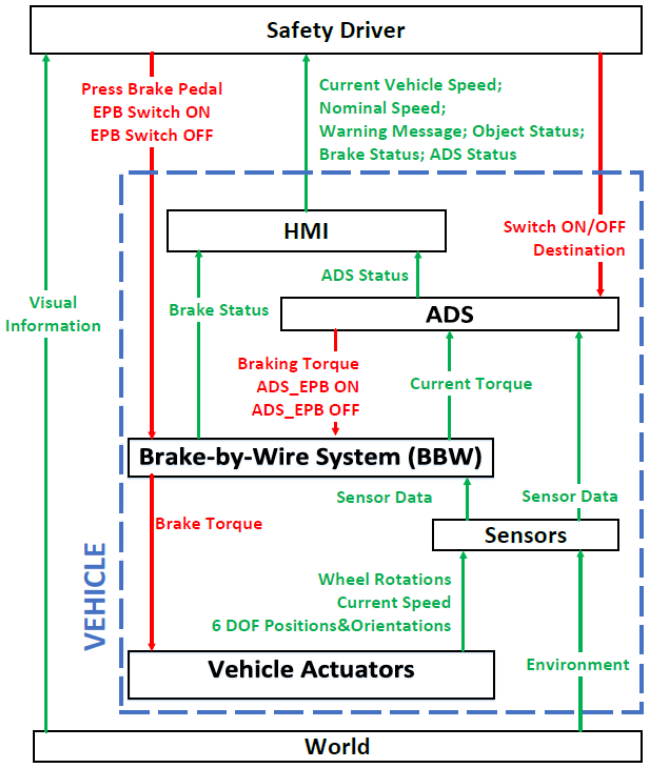

Fig. 3 A High Level Control Structure of the System under analysis

In UCA-1, Safety Driver does not press brake pedal when the vehicle speed is already over the speed limit, this fails to comply with the driving law (as per H-4). Considering UCA-2, when vehicle speed is too fast for the sharp turn, although braking is applied, the amount is insufficient and it is very likely that the vehicle overshoots the road (as per H-3) and therefore not able to follow the pre-defined route (as per H-2).

*D. **STPA Step 4:** Identify Loss Scenarios and Requirements*

Once the UCAs are identified, possible loss scenarios of the UCAs are identified. For each UCA, there are many possible loss scenarios. In this section, two example UCAs will be discussed further and one loss scenario of each example UCA will be identified.

1) Example UCA: Safety Driver did not press Brake Pedal

Considering **UCA-1** from TABLE III: Safety Driver does not press brake pedal when vehicle speed is higher than nominal speed (H-4).

Fig. 4 shows the control loop for UCA from Safety Driver, together with a human controller model embedded. In order to identify the loss scenarios for the UCA, we first need to understand the reasons behind the choice of the control action by the human operator:

- Safety Driver does not press brake pedal because of his driving skill. **(C-1)**
- Safety Driver decides not to press brake pedal because he has to follow the rules. **(C-2)**
- Safety Driver decides not to press brake pedal because of his knowledge of the vehicle system. **(C-3)**

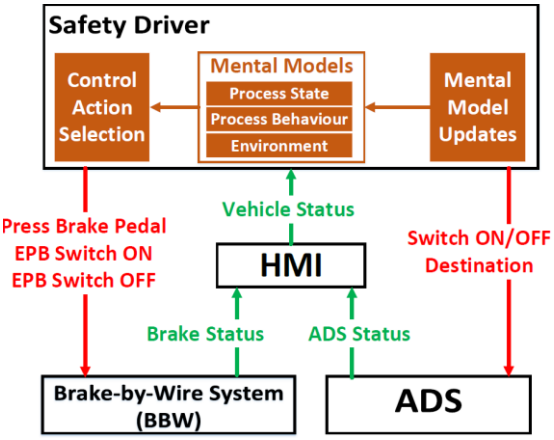

Fig. 4 Control Loop for the UCA from Safety Driver

Secondly, we also need to know what the human operator knows or believes about the system based on each of the reasons above. Considering knowledge-based decision **(C-3)**:

- Safety Driver was believing that vehicle speed was still lower than nominal speed. **(B-3.1)**
- Safety Driver was believing that the nominal speed did not change. **(B-3.2)**
- Safety Driver was believing that ADS was operating and would slow down vehicle automatically. **(B-3.3)**

And lastly, we need to identify how the operator comes to have their current knowledge or beliefs. Considering **(B-3.3)**:

- ADS status on HMI is incorrectly displayed. **(CF-3.3.1)**
- ADS status on HMI is not updated properly. **(CF-3.3.2)**
- ADS switch button performs multiple functions, which confuses the driver. **(CF-3.3.3)**

Loss scenario with Causal Factor **(CF-3.3.2)** can be mapped in the Human Controller Model as shown in Fig. 5. The Human Controller Model describes how and why the safety did not press brake pedal. Safety Driver decided not to press brake pedal even though he was aware that vehicle was overspeed. The decision

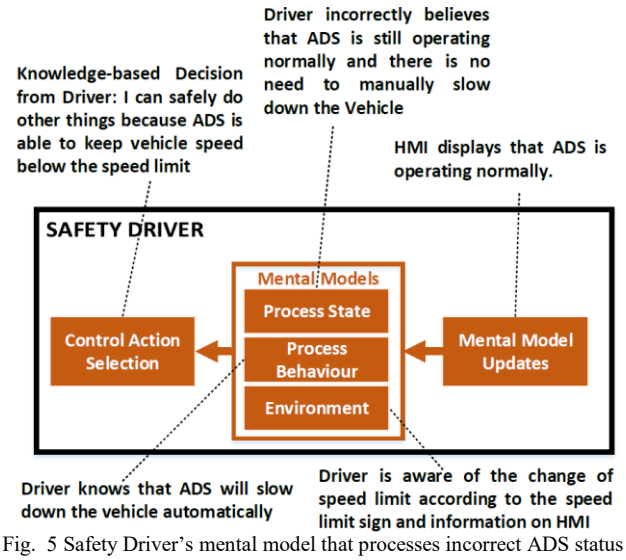

Fig. 5 Safety Driver's mental model that processes incorrect ADS status

was made based on his knowledge and beliefs about the ADS status. With the identification of the incorrect ADS status being displayed, the liability of the overspeed becomes convincing. In order to avoid the consequences of **(CF-3.3.2)**, two requirements are proposed below:

- **Requirement to prevent (CF-3.3.2):** When ADS is disabled, HMI shall update the ADS status immediately to inform driver. **(R-1)**
- **Requirement to detect (CF-3.3.2):** When ADS status is not properly displayed on HMI, warning messages shall be displayed to inform the driver. **(R-2)**

2) Example UCA: Safety Driver pressed Brake Pedal

Considering **UCA-2** from TABLE III: Safety Driver presses brake pedal with insufficient amount when current speed is still too high for the sharp turn ahead.

Same as the previous example, In order to identify the loss scenarios for the UCA, we first need to understand the reasons behind the choice of the control action by the human operator:

- Safety Driver did not press brake pedal with sufficient amount because of his driving skill. **(C-1)**
- Safety Driver did not press brake pedal with sufficient amount because he had to follow the rules. **(C-2)**
- Safety Driver did not press brake pedal with sufficient amount because of his knowledge of vehicle system. **(C-3)**

Secondly, we also need to know what the human operator knows or believes about the system based on each of the reasons above. Considering knowledge-based decision **(C-3)**:

- Safety Driver was believing that vehicle has already reached safe speed for the sharp turn. **(B-3.1)**
- Safety Driver was believing that the turn was not that sharp. **(B-3.2)**
- Safety Driver was believing that ADS was operating and would help with the deceleration. **(B-3.3)**

And lastly, we need to identify how the operator comes to have their current knowledge or beliefs. Considering **(B-3.2)**:

- The in-vehicle navigation map displays incorrect road structure. **(CF-3.2.1)**
- The in-vehicle navigation map is not updated properly. **(CF-3.2.2)**
- The vehicle body design increases Safety Driver's blind spot in terms of the surroundings. **(CF-3.2.3)**

Fig. 6 shows the Safety Driver's mental model when processing Causal Factor **(CF-3.2.1)**. Safety Driver was not aware of sharp road bend ahead due to blind spot. As in-vehicle map was the only information of forward road structure Safety Driver was receiving, he was trusting it although it was incorrectly displaying a smooth bend ahead. As a result, Safety Driver did not press brake pedal deeply enough, bringing the vehicle into an unsafe condition. In order to avoid consequences of **(CF-3.2.1)**, two requirements are proposed below:

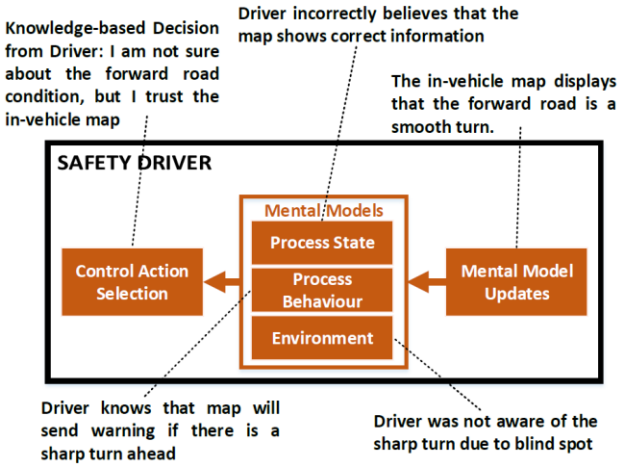

Fig. 6 Safety Driver's mental model that processes incorrect Map Information

- **Requirement to prevent (CF-3.2.1):** In-vehicle navigation map shall always be synchronized with real-world roads. **(R-1)**
- **Requirement to detect (CF-3.3.2):** When there is a mismatch between actual road and map, HMI shall inform driver. **(R-2)**

IV. Discussions

Two example UCAs have been discussed to demonstrate the applicability of STPA methodology on human-automation interactions. Considering safety driver's mental model in example UCA-2, in an ideal condition, a safety driver shall be well-trained with sufficient knowledge of controlling the vehicle and its automation system. Safety drivers select control actions based on their beliefs on the process state that information provided by vehicle systems are always correct, the process behaviour that map system will warn them if something is not right, and the surrounding environment they can see. When the information is inadequate or not updated properly, the mental model is updated with incorrect inputs, and therefore control actions that have been safe might become unsafe, i.e. UCA. As identified in the loss scenario of UCA-2, safety driver was not aware of the current environment (i.e. sharp turn) due to the blind spot, the control action (insufficient braking torque) that was believed to be safe was implemented by the safety driver, which was based on the information from in-vehicle map and the driver's trust in the capability of vehicle system. However, due to the causal factor that incorrect information was displayed on the map (i.e. CF-3.2.1), the knowledge-based actions (C-3) that safety driver believed were correct became unsafe (i.e. UCA-2), which potentially could lead to the hazards of collision with oncoming vehicles (i.e. H-1) or leaving pre-defined route (i.e. H-2). Furthermore, this could also affect safety driver's belief on the vehicle automation system on a long term basis, in which case safety driver might choose not to rely on vehicle automation systems at all (i.e. L-4). This violates the original intentions of designers, engineers and other stakeholders.

To ensure that the original intentions of building the system are not violated, the system design can be improved to ensure that the information is always correct (i.e. R-1). The possibility of UCAs can also be minimized if safety driver is informed of the abnormal system behaviour (i.e. R-2), in which case safety

driver is confident to make skill-based decisions to control the vehicle manually. Rather than blaming the driver for being the cause of the human error, system designers need to consider the factors influencing driver's mental models. This includes the consideration of how much effort the operator would require to access necessary information of the system. For example, an elaborate user manual handbook for the multi-functional vehicle control system covers all the necessary information, but it might not be that user friendly. As a result, most of drivers decide not to read through the handbook before starting their driving experiences with the vehicle.

Traditionally, if an accident occurs, analysts usually identify what the driver could have or should have done to prevent it [19]. Applying STPA on human-automation interactions allows analysts to identify possible combinations of human operators' beliefs on current process state, process behaviour and environment, together with their mental model updates. This helps analysts identify causes of accidents that were unknown. This could trace back to the difficulties associated with communications between vehicle HMI and drivers, the complexity of control modes available for drivers, the accuracy of the vehicle system, and even the inadequate decisions made by the vehicle manufacturing company who treats profits as a higher priority rather than safety. In the meantime, applying STPA also helps analysts identify what other control actions human operators could provide. As the system under analysis is still in the development phase, STPA as an prospective hazard identification method helps optimize the system in safety aspect as well as saving payments of potential accidents due to current system design flaws.

V. Conclusions

Process of STPA methodology has been discussed in this paper, along with two examples focusing on safety driver behaviour to demonstrate the applicability of STPA in identifying accidents relating to human-automation interaction. In the complete STPA analysis on the subject Vehicle System, 80 man-hours were spent and in total 18 Control Actions were identified, from which 205 UCAs were derived. This includes 37 UCAs from Safety Driver. From these 205 UCAs, 1850 causal factors were captured with 233 causal factors identified from Safety Driver UCAs. Among the UCAs and loss scenarios identified there is some fraction of unknown hazardous cases. Applying STPA also allows requirements to be generated directly and 3700 requirements have been proposed in this analysis to optimize the safety aspect of the system as well as the user interface design [21]. Part of the requirements will also be used as training instructions for Safety Drivers.

As part of the future work, it is important to verify the effectiveness of the analysis by testing. STPA as a hazardous scenario identification tool can be further extended for testing purposes. Building on Hazard Based Testing approach [6], STPA inspired test scenarios and test cases can be identified by parameterizing UCAs and loss scenarios to acquire test scenario parameters and pass criteria for testing in both real world and simulation world [21]. In the meantime, existing process still requires manual inputs to identify UCAs, and the quality and coverage of the analysis are therefore dependent on the knowledge and experience of the analysts. Therefore, future work is also needed to formalize the process of identifying UCAs to capture more unknown hazards.

Acknowledgement

The work presented in this paper has been carried under the Innovate UK and Centre for Connected and Autonomous Vehicles (CCAV) funded OmniCAV project. The authors would like to thank the WMG centre of HVM Catapult and WMG, University of Warwick, UK, for providing the necessary infrastructure for conducting this study. WMG hosts one of the seven centres that together comprise the High Value Manufacturing Catapult in the UK.

References

[1] R. N. Charette, "This car runs on code," IEEE Spectrum, vol. 46, no. 3, 2009.

[2] S. Khastgir, S. Birrell, G. Dhadyalla, and P. Jennings, "Identifying a gap in existing validation methodologies for intelligent automotive systems: Introducing the 3xD simulator," in Proc. of the IEEE Intelligent Vehicles Symposium 2015, 2015, pp. 648–653.

[3] S. Khastgir, S. Birrell, G. Dhadyalla, and P. Jennings, "Calibrating Trust to Increase the Use of Automated Systems in a Vehicle," in Advances in Human Aspects of Transportation. Advances in Intelligent Systems and Computing, vol 484, N. Stanton, S. Landry, G. Di Bucchianico, and A. Vallicelli, Eds. Springer, Cham, 2017, pp. 535–546.

[4] NHTSA, "Investigation Report: PE 16-007 (MY2014-2016 Tesla Model S and Model X)," 2017.

[5] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?," Transp. Res. Part A Policy Pract., vol. 94, no. December, pp. 182–193, 2016, doi: 10.1016/j.tra.2016.09.010.

[6] S. Khastgir, S. Birrell, G. Dhadyalla, and P. Jennings, "The Science of Testing: An Automotive Perspective," in SAE Technical Paper: 2018-01-1070, 2018, doi: 10.4271/2018-01-1070.

[7] Gary G. Kelm, "Failure Modes and Effects Analysis (FMEA), Critical Items List (CIL), and Fault Tree Analysis (FTA)," 2010.

[8] H. A. Duckworth and R. A. Moore, "Social responsibility: Failure mode effects and analysis," Soc. Responsib. Fail. Mode Eff. Anal., pp. 1–185, 2010, doi: 10.1201/EBK1439803721.

[9] W. E. Vesely and N. H. Roberts, Fault Tree Handbook. 1981.

[10] B. Kaiser, P. Liggesmeyer, and O. Mäckel, "A New Component Concept for Fault Trees," Proc. 8th Aust. Work. Saf. Crit. Syst. Softw., vol. 33, no. January, pp. 37–46, 2003.

[11] CENELEC, "Hazard and operability studies (HAZOP studies) — Application guide - EN 61882," 2016.

[12] G. H. Peng et al., "Risk Management in a Dynamic Society - A Modelling Problem," Chinese J. Schistosomiasis Control, vol. 30, no. 2, pp. 183–213, 2018, doi: 10.16250/j.32.1374.2016270.

[13] E. Hollnagel and Ö. Goteman, "The Functional Resonance Accident Model," Proc. Cogn. Syst. Eng. Process plant, pp. 155–161, 2004.

[14] S. A. Shappell and D. A. Wiegmann, "Applying Reason: The human factors analysis and classification system (HFACS)," Hum. Factors Aerosp. Saf. An Int. J. No.1, vol. 1, pp. 59–86, 2017.

[15] Eurocontrol, "A Method for Predicting Human Error in ATM (HERA-PREDICT)," 2004.

[16] N. G. Leveson, Engineering a Safer World. The MIT Press, 2011.

[17] N. Dulac, P. Nancy, L. Pi, and J. Laracy, "Demonstration of a New Dynamic Approach to Risk Analysis for NASA ' s Constellation Program," Mit, no. March, 2007.

[18] S. Khastgir, "Testing Automated Driving Systems To Calibrate Drivers ' Trust," no. May, 2019.

[19] M. E. France, "Engineering for Humans : A New Extension to STPA," p. 92, 2017.

[20] J. Thomas, "Extending and automating a systems-theoretic hazard analysis for requirements generation and analysis," Ph.D. Thesis, p. 232, 2013, doi: 10.2172/1044959.

[21] B. Gangopadhyay, S. Khastgir, S. Dey, P. Dasgupta, G. Montana, and P. Jennings, "Identification of Test Cases for Automated Driving Systems Using Bayesian Optimization," pp. 1961–1967, 2019.

[22] D. Rumsfeld, "Known and Unknown - A Memoir," Sentinel, p. 300, 2011.